

ARMY RESEARCH LABORATORY



A Fracturing of Polygonal Objects

by Andrew M. Neiderer, Mark A. Thomas,
and Richard Pearson

ARL-TR-1649

April 1998

19980805 025

Approved for public release; distribution is unlimited.

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

Citation of manufacturer's or trade names does not constitute an official endorsement or approval of the use thereof.

Destroy this report when it is no longer needed. Do not return it to the originator.

Army Research Laboratory

Aberdeen Proving Ground, MD 21005-5067

ARL-TR-1649

April 1998

A Fracturing of Polygonal Objects

Andrew M. Neiderer, Mark A. Thomas, Richard Pearson
Information Science and Technology Directorate, ARL

Approved for public release; distribution is unlimited.

Abstract

This report presents an approach for the fracture of polygonal objects. The technique was developed by the U.S. Army Research Laboratory (ARL) for real-time display of munitions penetrating urban structures. A weapons effects model and exterior polygon clipper were designed to accomplish this. The algorithm has been successfully implemented as a component of a combat simulator.

Acknowledgments

The authors of this report would like to thank Timothy Gion, a student at The University of Delaware, for his invaluable programming support. He was responsible for implementing this work on a Silicon Graphics, Inc., ONYX Reality Engine using Performer for real-time display.

INTENTIONALLY LEFT BLANK.

Table of Contents

	<u>Page</u>
Acknowledgments	iii
List of Figures	vii
1. Introduction	1
2. Fracture of a Polygonal Object	1
2.1 Circular Hole Formation	1
2.1.1 <i>Algorithm</i>	2
2.1.2 <i>Clipping of a Polygon Exterior to a Region</i>	2
2.2 Elliptical Hole Formation	4
2.2.1 <i>A True Elliptical Hole</i>	4
2.2.2 <i>A Jagged Elliptical Hole</i>	5
3. Real-Time Implementation	7
3.1 Penetration Mechanics	7
3.2 The Hole Library	8
4. Results and Future Work	9
5. References	11
Appendix	13
Distribution List	17
Report Documentation Page	19

INTENTIONALLY LEFT BLANK.

List of Figures

<u>Figure</u>	<u>Page</u>
1. A “Circular” (5^0 Increments for $0 \leq \delta < 90$) Hole	3
2. Exterior Polygon Clip of a Triangle	4
3. Hole Formation for an Origin-Centered Ellipse	5
4. An Ellipse Described by Five Clipping Rectangles	6
5. Elliptical Fracture of a Polygon for $\hat{v}_p = [-0.5777 \ 0.5777 \ 0.577]$	6
6. Flowchart for Object Damage Assessment	9
A-1. Perforated Urban Structure for Small Δ and $\delta = 22.5^0$	15

INTENTIONALLY LEFT BLANK.

1. Introduction

In an effort to better prepare soldiers in a battlefield environment, the U.S. Army Research Laboratory (ARL) is developing a comprehensive combat simulator [1]. Many components, such as dynamic terrain and battlefield smoke, have already been included. Recently, a capability for the approximation of a munition penetrating an urban structure was added. In order to maintain real-time display of this phenomenon, weapons effects had to be simplified.

Weapons effects can be divided into blast and kinetic energy (KE) effects. A blast weapon detonates its high explosive (HE) warhead at the target surface or inside the structure. This usually results in extensive damage; large HE warheads can even cause structural collapse. However, this paper only addresses KE effects. A solid KE penetrator delivers energy to a localized region of the target, and typically results in a small perforation with respect to the size of the impacted object.

The following sections present a technique for geometrical change and attribute update of a polygonal object that undergoes fracture. The algorithm can handle penetrations in any part of a polygonal object. This includes the ideal situation that results in a circular hole, as well as the more general case of elliptical hole formation. A simple empirical solution for more realistic perforations is then described. Finally, ARL's implementation of the method is discussed, and an example is given.

2. Fracture of a Polygonal Object

The penetrating projectile p is assumed to have a circular, cross-sectional area of radius r_p ; the target t is a polygonal object, where each surface is defined by coplanar vertices. These two restrictions allow for the application of polygon clipping.

Polygon clipping is a procedure for determining which portions of a subject polygon lie inside and/or outside a clipping region. Typically, the algorithm starts by clipping a polygon against a single, infinite clip edge (2D) or plane (3D). If there is no intersection, the polygon is either trivially accepted or rejected. Otherwise, the points where polygon edges intersect the clipping region are computed. These intersection points are vertices of a new intermediate polygon that is then clipped against the next clipping region and so on. Note that polygons must remain closed (i.e., both input and output to a polygon clipper are polygons).

The next section first presents an algorithm for the generation of a circular hole in a target, then a solution for the more general case, where the normal direction vector of the target \hat{u}_t , does not necessarily coincide velocity vector of the projectile \vec{v}_p , is described. Both involve geometrical manipulation of objects in the 3D scene and a subsequent application of an exterior polygon clipper.

2.1. Circular Hole Formation

The following algorithm could be used for a normal impact, where the direction of the projectile velocity vector aligns with the unit normal of the target. The geometry in this case is simpler than that for an oblique impact and allows one to focus on the process of hole formation.

2.1.1. Algorithm

Every object in the 3D scene is described by one or more planar polygons. Each of these polygons is positively oriented (i.e., the order of vertices is counterclockwise as you face the polygon). Because polygon vertices are ordered, an edge can be treated as a vector and therefore the normal vector \vec{n} to the plane of a polygon is computed by taking the cross product of adjacent edge vectors; the unit normal \hat{n} is $\vec{n} / |\vec{n}|$.

Once the point of impact (x_i, y_i, z_i) is known, each polygon of the affected object is geometrically transformed so that (x_i, y_i, z_i) is at the origin, and the unit normal \hat{n} is aligned with the positive z -axis.¹ This is accomplished by first rotating about the x -axis through an angle α , and then about the y -axis by β degrees [2]. An exterior polygon clip is then performed about a region containing (x_i, y_i) , where $z_i = 0$ for each rotation δ about the z -axis. Thus, the complete procedure is to

- (1) translate the object so that (x_i, y_i, z_i) is at the origin,
- (2) rotate about the x -axis till \hat{n} lies in the xz plane,
- (3) rotate about the y -axis to make \hat{n} coincident with the z -axis,
- (4) for each rotation δ about the z -axis perform an exterior polygon clip,
- (5) apply the inverse of (3),
- (6) apply the inverse of (2), and
- (7) apply the inverse of (1).

If the point of impact is designated as the centroid of a 2D clipping square, then performing an exterior polygon clip for $0 \leq \delta < 90$ at small increments results in a “circular” hole (see Figure 1). The details of an exterior polygon clip are now examined.

2.1.2. Clipping a Polygon Exterior to a Region

There are many published solutions for clipping polygons (convex or concave) interior to a region. We have modified Heckbert’s implementation of the Sutherland-Hodgman (SH) algorithm [3] for exterior polygon clipping. Heckbert uses an interpolation mask for the efficient update of polygon vertex attributes; relevant attributes are bilinearly interpolated across the face of the polygon. However, the code has two restrictions: (1) subject polygons must be convex, and (2) the clipping region be defined as a 3D rectangular parallelepiped.

A subject polygon from the appropriate object is passed through the modified SH polygon clipper. A trivial accept/reject is first determined by counting vertices outside/inside with respect to each of the four clip edges. If the subject polygon is completely outside each of the four edges, it does not have to be clipped. If it is entirely within each of the four edges it is discarded. Only if neither of these conditions applies (i.e., crosses an edge) does it have to be clipped.

The subject polygon, which is convex, is clipped against one of the four infinite clip edges. The resulting exterior polygon, if any, will also be convex. This remainder polygon is then clipped against each of the other three clip edges in succession.

¹The choice of the z -axis is arbitrary. The idea is to make \hat{n} coincide with one of the coordinate axes, which will then be the rotation axis.

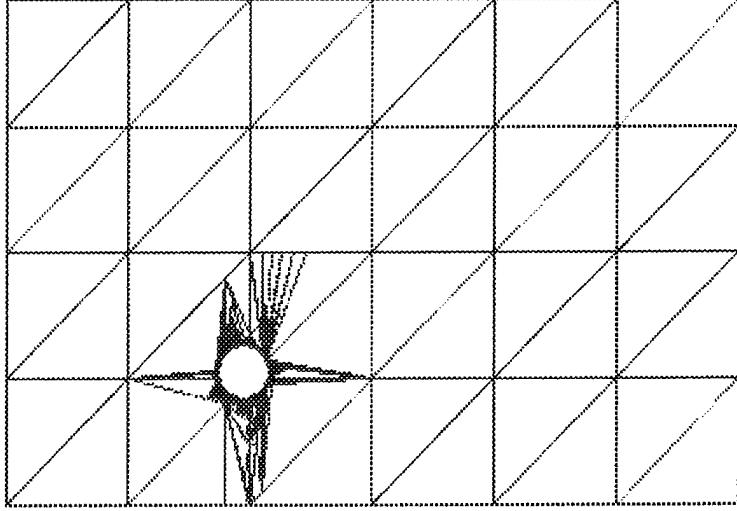


Figure 1: A "circular" (5^0 increments for $0 \leq \delta < 90$) hole.

Figure 2 shows an example of a nontrivial exterior polygon clip. The planar polygon consists of 48 subject polygons (Figure 2a). As shown in Figure 2b, only polygons 19 and 20 need to be clipped against the clipping square (i.e. there are 46 trivial cases). Let us examine the exterior polygon clip of polygon 19 (Figure 2c). Clipping this polygon against the left, infinite clip edge results in an exterior polygon and an intermediate polygon (Figure 2d). In Figure 2e, the intermediate polygon is clipped to the right clip edge, which results in a second exterior polygon. The new intermediate polygon does not have to be clipped against the bottom clip edge since there is no intersection (Figure 2f). Finally, the intermediate polygon is clipped against the top clip edge (Figure 2g). The three resultant exterior polygons are saved and could be either output or used as input for a new orientation of the clipping square.

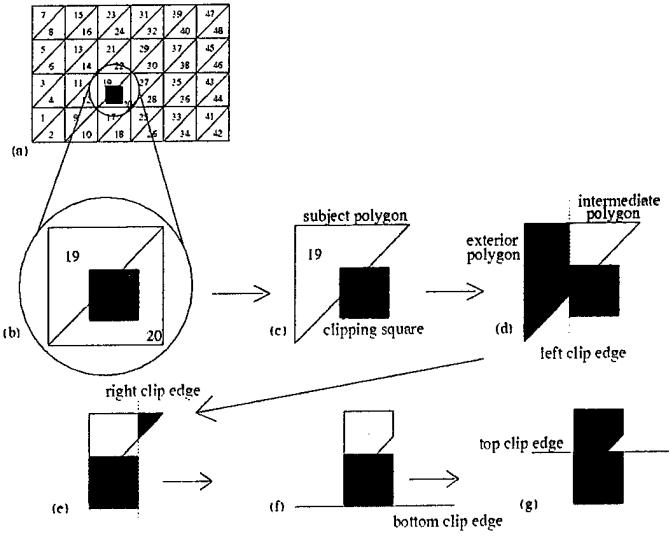


Figure 2: Exterior polygon clip of a triangle.

2.2. Elliptical Hole Formation

This section considers the geometry for a projectile penetrating at an oblique angle. A simple solution for realistic perforations of a target is presented.

2.2.1. A True Elliptical Hole

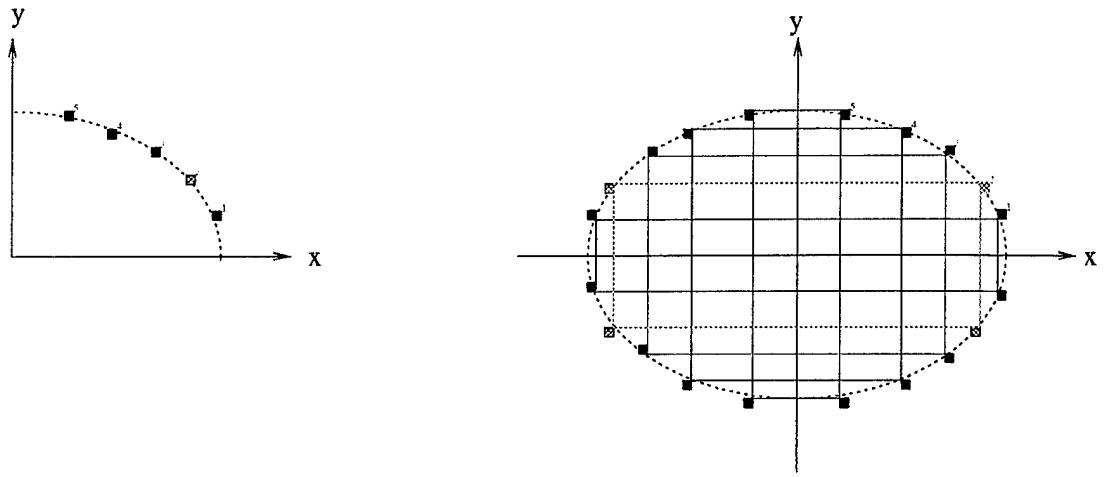
A penetrating projectile of radius r_p results in a conic (minus the hyperbola) perforation of the target. The shape is a function of $\cos(\gamma)$, where γ is the angle between \vec{v}_p and \hat{u}_t satisfying $0 \leq \gamma < 90$. The upper limit of γ is the grazing angle. A $\gamma = 0$ corresponds to normal impact and forms a circular perforation. Any other value of γ results in an elliptical shape.

The true elliptical hole has semiminor axis length $b = r_p$ and semimajor axis length $a = r_p / \cos(\gamma)$. The first 3 steps of the algorithm presented in section 2.1.1 place the point of impact at the origin with the unit normal of the appropriate polygon aligned with the positive z -axis. The next step is to rotate about the z -axis by some angle η , aligning the semimajor axis with the x -axis at the plane of the target. The inverse of these four steps needs to be applied after exterior polygon clipping at this orientation.

An aligned, origin-centered ellipse can be mathematically described by the parametric functions

$$\begin{aligned} x(\theta) &= a\cos(\theta), \text{ and} \\ y(\theta) &= b\sin(\theta) \end{aligned}$$

for $0 \leq \theta < 2\pi$ radians. The 2D coordinates $[x(\theta) \ y(\theta)]$ of points along the perimeter of the ellipse in Euclidean space are computed by taking uniform angular increments of the parameter θ . However, one can decrease computation time by restricting the range to $0 \leq \theta \leq \pi/2$, and then use four-way symmetry (4WS) to determine the rest of the points.



(a) first quadrant ellipse showing vertices
of 5 clipping rectangles

(b) clipping rectangles as determined by 4-way symmetry

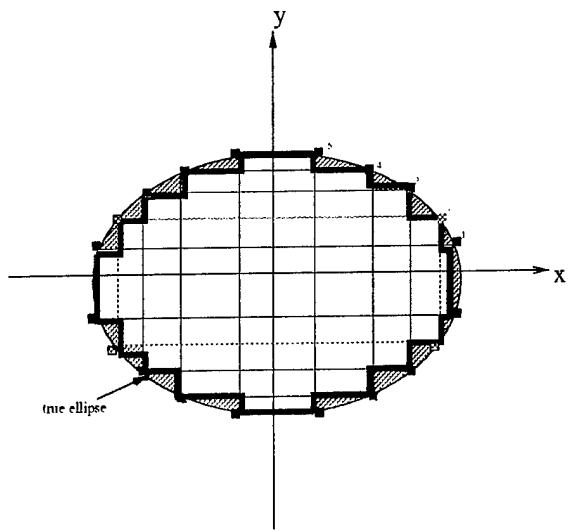
Figure 3: Hole formation for an origin-centered ellipse.

Figure 3 illustrates the previously mentioned technique. The points satisfying $0 < \theta < \pi/2$ at an angular increment $d\theta = \pi/12$ are shown in Figure 3a; the upper and lower limit, namely $\theta = 0$ and $\theta = \pi/2$, are not needed when determining clipping rectangles for elliptical hole formation. Symmetrical points and clipping rectangles are then determined (Figure 3b).

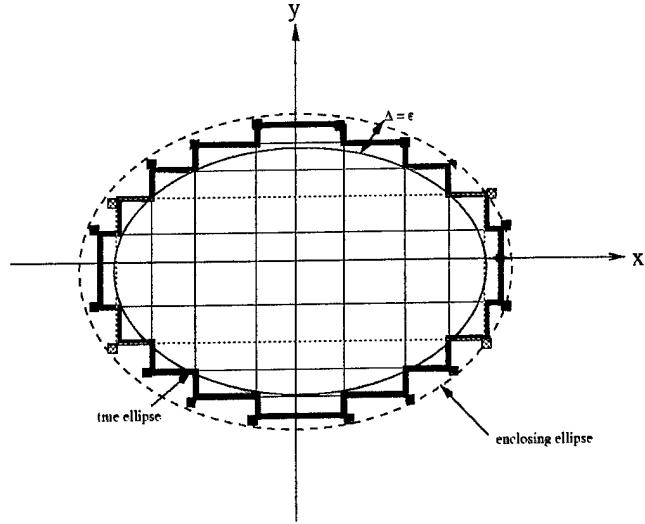
2.2.2. A Jagged Elliptical Hole

An infinite number of clipping rectangles n , where n is a natural number, are needed to generate a true elliptical hole using this approach, and the number of computations increases as $n \rightarrow \infty$. Decreasing n reduces computation time, but may result in unacceptable error. Consider Figure 4, where $n = 5$. An exterior polygon clip against any of these clipping rectangles could result in a vertex within the shaded region (see Figure 4a). This is not physically possible since the hole must be at least the size of the true ellipse. No matter how large of an n is used, there will still be some associated error. Therefore, the exterior polygon clip needs to be done against circumscribed clipping rectangles (see Figure 4b). In this figure ϵ corresponds to the distance for a minimum enclosing band. A uniform random number generator (RNG) is used to add noise to position vectors of polygon vertices that are within some Δ of the true ellipse. This technique provides for a more realistic-looking perforation.

An example of elliptical fracture using five clipping rectangles is given in Figure 5. The impact point for the penetrating projectile of velocity direction vector $[-0.577 \ 0.577 \ 0.577]$ is located at the center of the $z = 0$ planar polygon.

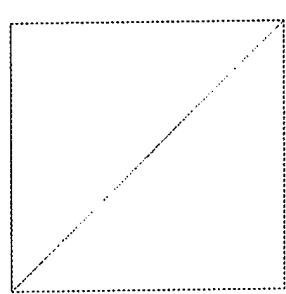


(a) inscribed clipping rectangles

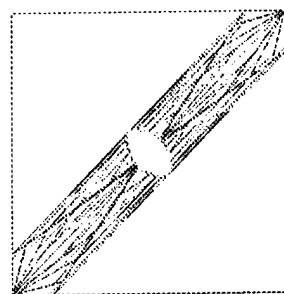


(b) circumscribed clipping rectangles

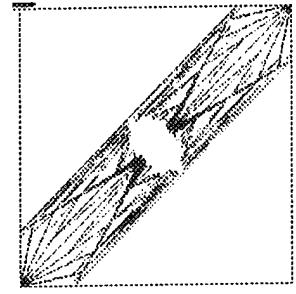
Figure 4: An ellipse described by five clipping rectangles.



(a) polygon consisting of two triangles



(b) elliptical hole



(c) jagged elliptical hole for small Δ

Figure 5: Elliptical fracture of a polygon for $\hat{v}_p = [-0.577 \ 0.577 \ 0.577]$.

3. Real-Time Implementation

The fracture algorithm has been integrated into a Distributed Interactive Simulation (DIS). The following discusses the implementation for ARL's Stealth DIS simulator. This involved the development of a modular library libhole; but before using this library, a determination is made as to whether penetration even occurs.

3.1. Penetration Mechanics

The Detonation Protocol Data Unit (DPDU) and the Collision Protocol Data Unit (CPDU) contain properties of the projectile and target. The projectile has a circular, cross-sectional area of radius r_p . It is also characterized by its mass m_p and a velocity vector \vec{v}_p . The target is homogeneous and isotropic throughout its thickness t_t . Its orientation is specified by the normal direction vector \hat{u}_t . The penetration depth z is computed by using one of the following empirical relationships [4].

(1) For a ceramic target,

$$z_c = \frac{222 * p_p * d_p^{0.215} * |\vec{v}_p|^{1.5}}{\sigma_t^{0.5}} + 0.5 * d_p, \text{ where}$$

p_p is the sectional pressure of the projectile (lb/in^2),

d_p is the diameter of the projectile (in),

$|\vec{v}_p|$ is the velocity of the projectile ($\text{ft/s} \times 1000$), and

σ_t is the compressional strength of the ceramic target (lb/in^2).

(2) For a metal target,

$$z_0 = 0.21 * w_p^{0.33} * |\vec{v}_p|^{1.22} \text{ and}$$

$$z_m = z_0 * e^{(8.77 * 10^{-6} * (B_t^2 - B^2)) - (5.41 * 10^{-3} * (B_t - B))}, \text{ where}$$

w_p is the weight of the projectile (oz),

$|\vec{v}_p|$ is the velocity of the projectile ($\text{ft/s} \times 1000$),

B_t is the Brinell hardness of the metal target, and

B is the Brinell hardness of common mild structural steel (i.e., $B = 150$).

(3) For a wood target,

$$z_w = \frac{10041 * |\vec{v}_p|^{0.41} * w_p^{0.59}}{\rho_t * h_t^{0.54}}, \text{ where}$$

w_p is the weight of the projectile (oz),

$|\vec{v}_p|$ is the velocity of the projectile ($\text{ft/s} \times 1000$),

ρ_t is the density of the wood target (lb/ft^3), and
 h_t is the hardness of the wood target (lbs).

The simulator then determines the geometry at the point of impact by executing the Performer² function pfNodeIsectSegs(). The data is stored in the following structure :

```
typdef struct {
    long flags;
    pfVec3 point;
    pfVec3 norm;
    long tri;
    long prim;
    pfGeoSet *gset;
    pfGeode *geode;
    pfMatrix mat;
} HitInfo.
```

The fields include the point of intersection, the normal vector at this point, identification number of the triangle that was impacted, the number of the corresponding primitive, and various geometry parameters.

The penetration depth z for the impacted object is compared to the line-of-sight thickness.

$$t_{LOS} = t_t / \cos(\gamma),$$

where γ is the angle between the direction vector of the projectile and the unit normal of the target satisfying $0 \leq \gamma < 90$. If $z > t_{LOS}$, then there is penetration and the hole() function is called (see Figure 6).

3.2. The Hole Library

The libhole library consists of the simulation interface module (SIM) and the polygon clipper module (PCM). The polygon clipper, which was described in section 2, produces the actual geometry modifications. The SIM generates the appropriate data structures needed by the PCM, and converts the output of the polygon clipper into simulation geometry; it uses Performer pfdBuilder functions to create this resultant geometry.

The calling syntax is

```
hole(pfGroup *parent, HitInfo *hitinfo, float diameter, float gamma).
```

The parent geometry node may be passed as NULL. The HitInfo structure points to the original geometry set data structure, or pfGeoSet; this contains the actual vertex information needed by the PCM. The other two parameters define the shape of the perforation.

²Performer is a registered trade mark of Silicon Graphics, Inc. (SGI).

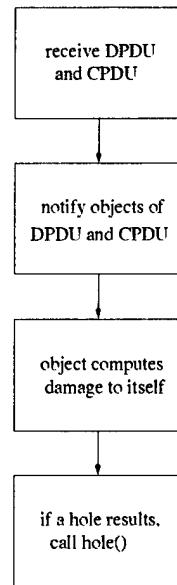


Figure 6: Flowchart for object damage assessment.

4. Results and Future Work

A combat simulator for the training of soldiers should include events that may influence decision-making within that environment. This paper addressed one such event: a munition that penetrates an urban structure, which involves fracture of a polygonal object in the ARL Stealth simulator environment. The approach taken should also allow for easy integration into existing simulators as well.

An example is given in the appendix. Here an explosive had been placed on the wall, then detonated for tactical entry to the building. Real-time simulation of this scenario has been demonstrated at the Association of the United States Army (AUSA) annual meeting held in Washington D.C. on October 1996.

Since an impact may result in a large number of polygons, both in the target and fragments, decimation logic to reduce the polygon count is being implemented. Future work should include a more comprehensive database of material types. In addition, the effect of multiple perforations against a structure, which could result in structural failure, needs to be investigated.

INTENTIONALLY LEFT BLANK.

References

1. Thomas, Mark A., "Interactive Simulation Software Testbed (STEALTH): Capabilities, Applications, and Software Description", U.S. Army Research Laboratory, Aberdeen Proving Ground, MD, 1996.
2. Rogers, David F. and Adams, J. Alan, *Mathematical Elements for Computer Graphics*, McGraw-Hill Publishing Company, 1990.
3. Glassner, Andrew S., ed., *Graphics Gems*, Academic Press, Cambridge, MA, 1990.
4. Department of the Army, "Fundamentals of Protective Design for Conventional Weapons", Technical Manual 5-855-1, Washington D.C., July 1988.

INTENTIONALLY LEFT BLANK.

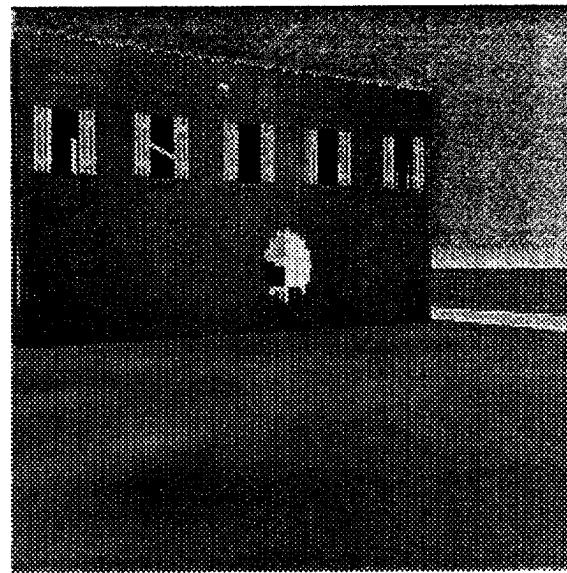
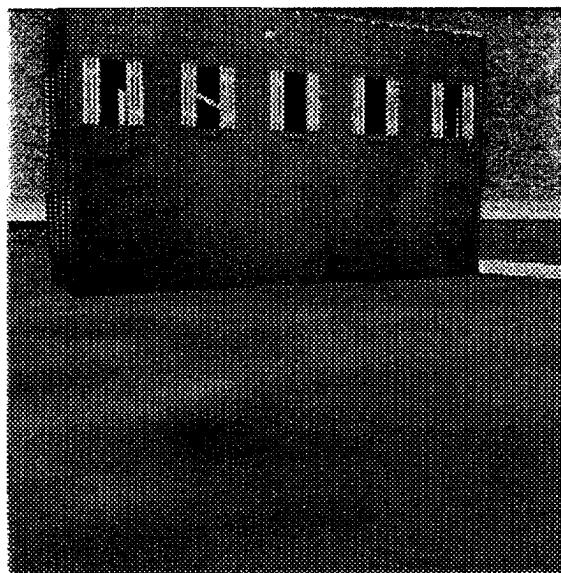
APPENDIX:

Wall Perforation

INTENTIONALLY LEFT BLANK.

Appendix

These two figures represent successful penetration of the wall of a building at normal incidence. Note the objects inside the building and that the “hole” is somewhat regular.



(a) before penetration

(b) after penetration

Figure A-1: Perforated urban structure for small Δ and $\delta = 22.5^\circ$.

INTENTIONALLY LEFT BLANK.

<u>NO. OF COPIES</u>	<u>ORGANIZATION</u>	<u>NO. OF COPIES</u>	<u>ORGANIZATION</u>
2	DEFENSE TECHNICAL INFORMATION CENTER DTIC DDA 8725 JOHN J KINGMAN RD STE 0944 FT BELVOIR VA 22060-6218	1	INST FOR ADVNCED TCHNLGY THE UNIV OF TEXAS AT AUSTIN PO BOX 202797 AUSTIN TX 78720-2797
1	HQDA DAMO FDQ DENNIS SCHMIDT 400 ARMY PENTAGON WASHINGTON DC 20310-0460	1	CECOM PM GPS COL S YOUNG FT MONMOUTH NJ 07703
1	DPTY ASSIST SCY FOR R&T SARD TT F MILTON RM 3EA79 THE PENTAGON WASHINGTON DC 20310-0103	1	GPS JOINT PROG OFC DIR COL J CLAY 2435 VELA WAY STE 1613 LOS ANGELES AFB CA 90245-5500
1	OSD OUSD(A&T)/ODDDR&E(R) J LUPO THE PENTAGON WASHINGTON DC 20301-7100	1	ELECTRONIC SYS DIV DIR CECOM RDEC J NIEMELA FT MONMOUTH NJ 07703
1	CECOM SP & TRRSTRL COMMCTN DIV AMSEL RD ST MC M H SOICHER FT MONMOUTH NJ 07703-5203	3	DARPA L STOTTS J PENNELLA B KASPAR 3701 N FAIRFAX DR ARLINGTON VA 22203-1714
1	PRIN DPTY FOR TCHNLGY HQ US ARMY MATCOM AMCDCG T M FISSETTE 5001 EISENHOWER AVE ALEXANDRIA VA 22333-0001	1	US MILITARY ACADEMY MATH SCI CTR OF EXCELLENCE DEPT OF MATHEMATICAL SCI MDN A MAJ DON ENGEN THAYER HALL WEST POINT NY 10996-1786
1	PRIN DPTY FOR ACQUSTN HQ US ARMY MATCOM AMCDCG A D ADAMS 5001 EISENHOWER AVE ALEXANDRIA VA 22333-0001	1	DIRECTOR US ARMY RESEARCH LAB AMSRL CS AL TP 2800 POWDER MILL RD ADELPHI MD 20783-1145
1	DPTY CG FOR RDE HQ US ARMY MATCOM AMCRD BG BEAUCHAMP 5001 EISENHOWER AVE ALEXANDRIA VA 22333-0001	1	DIRECTOR US ARMY RESEARCH LAB AMSRL CS AL TA 2800 POWDER MILL RD ADELPHI MD 20783-1145
		3	DIRECTOR US ARMY RESEARCH LAB AMSRL CI LL 2800 POWDER MILL RD ADELPHI MD 20783-1145

NO. OF
COPIES ORGANIZATION

1 USA STRICOM PM DIS
AMCPM DIS
CENTRAL FLORIDA RSRCH PK
12350 RSRCH PKY
ORLANDO FL 32826-3276

1 USA STRICOM
AMSTI ET TRACI JONES
CENTRAL FLORIDA RSRCH PK
12350 RSRCH PKY
ORLANDO FL 32826-3276

1 COMMANDANT
USIS
ATSH CDA JAN CHERVENAK
DISMOUNTED BATTLESPACE
BATTLE LAB
FORT BENNING GA 31905-5400

1 REALITY BY DESIGN
PAUL BARHAM
867 WAVE ST STE 200
MONTEREY CA 93940-1054

ABERDEEN PROVING GROUND

10 DIR USARL
AMSLR CI LP (305) (4 CPS)
AMSLR IS
J GANTT
P EMMERMAN
AMSLR IS C COL M KINDL
AMSLR IS OCD
M A THOMAS
A NEIDERER
AMSLR WM WE R PEARSON

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188
<p>Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project(0704-0188), Washington, DC 20503.</p>			
1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED	
	April 1998	Final, January-April 1997	
4. TITLE AND SUBTITLE		5. FUNDING NUMBERS	
A Fracturing of Polygonal Objects		PR: P622618.H80	
6. AUTHOR(S)			
Andrew M. Neiderer, Mark A. Thomas, and Richard Pearson			
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)		8. PERFORMING ORGANIZATION REPORT NUMBER	
U.S. Army Research Laboratory ATTN: AMSRL-IS-C Aberdeen Proving Ground, MD 21005-5067		ARL-TR-1649	
9. SPONSORING/MONITORING AGENCY NAMES(S) AND ADDRESS(ES)		10. SPONSORING/MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES			
12a. DISTRIBUTION/AVAILABILITY STATEMENT		12b. DISTRIBUTION CODE	
Approved for public release; distribution is unlimited.			
13. ABSTRACT (Maximum 200 words)			
<p>This report presents an approach for the fracture of polygonal objects. The technique was developed by the U.S. Army Research Laboratory (ARL) for real-time display of munitions penetrating urban structures. A weapons effects model and exterior polygon clipper were designed to accomplish this. The algorithm has been successfully implemented as a component of a combat simulator.</p>			
14. SUBJECT TERMS		15. NUMBER OF PAGES	
polygon clipping, mathematical elements of computer graphics, penetration mechanics, dynamic terrain		20	
		16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT		18. SECURITY CLASSIFICATION OF THIS PAGE	
UNCLASSIFIED		UNCLASSIFIED	
19. SECURITY CLASSIFICATION OF ABSTRACT		20. LIMITATION OF ABSTRACT	
UNCLASSIFIED		UL	

INTENTIONALLY LEFT BLANK.

USER EVALUATION SHEET/CHANGE OF ADDRESS

This Laboratory undertakes a continuing effort to improve the quality of the reports it publishes. Your comments/answers to the items/questions below will aid us in our efforts.

1. ARL Report Number/Author ARL-TR-1649 (Neiderer) Date of Report April 1998

2. Date Report Received _____

3. Does this report satisfy a need? (Comment on purpose, related project, or other area of interest for which the report will be used.) _____

4. Specifically, how is the report being used? (Information source, design data, procedure, source of ideas, etc.) _____

5. Has the information in this report led to any quantitative savings as far as man-hours or dollars saved, operating costs avoided, or efficiencies achieved, etc? If so, please elaborate. _____

6. General Comments. What do you think should be changed to improve future reports? (Indicate changes to organization, technical content, format, etc.) _____

Organization _____

CURRENT
ADDRESS

Name _____ E-mail Name _____

Street or P.O. Box No. _____

City, State, Zip Code _____

7. If indicating a Change of Address or Address Correction, please provide the Current or Correct address above and the Old or Incorrect address below.

Organization _____

OLD
ADDRESS

Name _____

Street or P.O. Box No. _____

City, State, Zip Code _____

(Remove this sheet, fold as indicated, tape closed, and mail.)
(DO NOT STAPLE)